

REALIZATION OF ARITHMETIC OPERATIONS ON SIGNAL SEQUENCES USING NORMAL ALGORITHMS AND REWRITING CYCLIC NORMAL AUTOMATON

U. M. FERNANDES DIMLO¹ & E. G. RAJAN²

¹Research Scholar, University of Mysore, Karnataka, India

²Founder & President, Pentagram Research Center Pvt. Ltd, Hyderabad, Andhra Pradesh, India

ABSTRACT

Symbolic computation is study and development of algorithms and software for manipulating mathematical expressions and objects. In signal processing data is represented using symbols. But today computers are basically numeric computers hence automation of basic and advanced signal processing operations became very difficult task. Even though effort has been made to write algorithms and programs for signal processing operations but time complexity of such implementations are very poor. In this paper signals are represented as strings or words which can be stored in computer memory, further normal algorithms are written to perform basic arithmetic operations like addition, difference, product and division of two signals using rewriting cyclic normal automata.

KEYWORDS: RCNA, Normal Algorithms, Substitution Formula

INTRODUCTION

Data can be represented either using numbers or symbols. Hence based on data representation there are two types of computation can be performed on data such as

- Numeric computation
- Symbolic computation

Computation carried out by human beings before invention of computer is purely symbolic. Once computers come into existence numeric computation became very popular. Any task which is solved by computers takes data in the [10] form of numbers. For example looping a certain task for n number of times, doing any bank transaction, finding integration using Simpson's method are carried out in a numeric way. But there is no significant improvement in numerical differentiation.

Symbolic computation is study and development of algorithms and software for manipulating mathematical expressions and objects. In signal processing data is represented using symbols. But today computers are basically numeric computers hence automation of basic and advanced signal processing operations became very difficult task. Even though effort has been made to write algorithms and programs for signal processing operations but time complexity of such implementations are very poor. The problems identified are processing signal processing operations [09] on numeric computers, difficulty in writing algorithms, Rare availability of software to perform advanced signal processing operations, Modeling symbolic operations in computer process able form, Poor time complexity of existing methods.

DEFINITION OF IMPORTANT TERMS

Rewriting Cyclic Normal Automata

Let us consider a normal algorithm N consisting of n totally ordered substitution formulas $\{u_i \rightarrow v_i; 1 \leq i \leq n\}$ and its EPT version [9]. We shall define a rewriting cyclic normal automaton for the EPT system in the following manner:

Let Q be the set of states corresponding to the semi-Thue productions [11] contained in the scheme of the EPT rewriting system where $Q = \{q_{ij} \mid 0 \leq i \leq n - 1; 0 \leq j \leq k; i \text{ represents the } i^{\text{th}} \text{ substitution formula, } j \text{ represents the } j^{\text{th}} \text{ semi-Thue production of the } i^{\text{th}} \text{ substitution formula [12] and } k \text{ represents the number of states, that is, the number of semi-Thue productions corresponding to the } i^{\text{th}} \text{ substitution formula}\}$. For convenience, the state corresponding to the i^{th} substitution formula is termed as a *major state* and the states corresponding to the k semi-Thue productions of the i^{th} substitution formula as *minor states*. [13] Let $u_i \rightarrow v_i$ be the i^{th} substitution formula. Then, the number of minor states corresponding to the i^{th} major state is given by the following rules.

- if $|u_i| = |v_i| = l_m$, then $k = 2l_m$

[NOTE: $|u_i|$ is the length of u_i]

- if $|u_i| < |v_i|$ then $k = 2l_m$
- if $|u_i| > |v_i|$ then $k = 2l_m - 1$

Substitution Formulas

The formula of the form $P \rightarrow Q$ used in normal algorithms to derive given string is called as substitution formula. There are two types of substitution formulas

- Simple substitution formula
- Terminal substitution formula

This formula comprises of generic variables, auxiliary [14] variables and string. Terminal formula will be used to end the derivation where simple formula used to replace part of a string by another string while derivation.

Normal Algorithms

Finite set of symbols are known as alphabet denoted by A or Σ . A word w is finite sequence of symbols from Σ . Basic set theory operations and notations like intersection, union, set difference, concatenation, subset, proper subset, belongs to, not belongs to can be the same in automata also. A variable is said to be generic [15] variable whose values are taken from alphabet Σ . Usually generic variables are denoted by symbols like ξ , η and μ . In automata null string is represented by the symbol Λ . A Markov Algorithm over an alphabet A is a finite ordered sequence of productions $x \rightarrow y$, where $x, y \in A^*$. Some productions may be "Halt" productions. e. g. $abc \rightarrow b$, $ba \rightarrow x$ (halt).

ADDITION OF TWO SIGNALS

Let s_1, s_2 be any two signals which represented using strings w_1, w_2 in automata. Because in automata everything which has been recognized will be considered as word. Let we convert given signals into integer sequence where for example integer 3 will be represented using ||| and integer 5 will be represented as |||||. Hence numbers of vertical lines are equal to given integer sequence of input signal.

Let N^{ADD} is an RCNA based normal algorithm which is used to add to signals denoted by integer sequence. The substitution formulas for addition are as follows.

Table 1: Formula for Addition Two Signals

Formula	Formula Number	Comments
$+ \rightarrow \Lambda$	0	Replace symbol + by null string and stop the addition process

Let us consider two signals which are represented as integer sequence $|||$ and $||$. We have only one formula in which + is replaced by Λ then if we concatenate any string with null string the result will be the same string. The addition of these two signals carried out as follows.

Table 2: Addition of Two Signals 3 and 2

Step No.	Transformation	Formula Used
1	$ + $	Input string
2	$ \Lambda $	Formula 0
3	$ $	Resultant string ($w \Lambda = \Lambda w = w$)

The RCNA for the above EPT system is [1,2] defined by $C^{ADD} = \langle Q, INI, FIN \rangle$ where $Q = \{ q_{00}, q_{10}, q_{20}, q_{21}, q_{30}, q_{40}, q_{50}, q_{51} \}$ with the set of edges T where, $T = \{ (q_{INI}, \Sigma^*, q_{00}), (q_{00}, \Sigma \backslash \xi / \leftarrow, q_{10}), (q_{10}, \Sigma \backslash \xi / \leftarrow, q_{20}), (q_{20}, \Sigma \backslash \xi / \leftarrow, q_{30}), (q_{20}, + / \Lambda, q_{21}),$

$(q_{21}, \Lambda / \vdash, q_{00}), (q_{30}, \Sigma \backslash \xi / \leftarrow, q_{40}), (q_{40}, \Sigma \backslash \xi / \leftarrow, q_{50}), (q_{50}, \Sigma \backslash \xi / \leftarrow, q_{FIN}), (q_{50}, \Lambda / \Lambda, q_{51}), (q_{51}, \Lambda / \vdash, q_{00}) \}$

The graphical representation of C^{ADD} is shown in figure 1

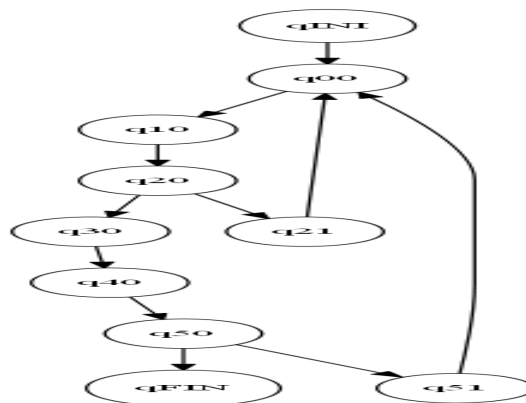


Figure 1: RCNA for Addition of Two Signals

DIFFERENCE OF TWO SIGNALS

The signals which are given will be converted into integer sequence $p(n)$ and $q(n)$ where n is length of the sequence. In general pipeline operator is used to denote signals in automata theory. In finding difference of two signals three cases may arise which are listed below

- Integer sequence of signal s_1 is greater than signal s_2 sequences
- Signal s_1 integer sequence is same as s_2 sequence

- Length of $p(n)$ is shorter than length of $q(n)$

Let N^{SUB} is an RCNA based normal algorithm which is used to add to signals denoted by integer sequence. The substitution formulas for addition are as follows.

Table 3: Substitution Formula for to Find Difference of Signals

Formula	Formula Number	Comments
$ - \rightarrow -$	0	Substitute $-$ in the place $ - $ in the given input string
$ - \rightarrow $	1	Replace at last $ -$ with $ $
$- \rightarrow 0$	2	If both signals are same result is 0

Let us consider two signals which are represented as integer sequence $|||$ and $||$. We should use formula 0 for twice then formula 1 or once to get desired result. The subtraction of these two signals carried out as follows.

Table 4: Difference of Signals 3 and 2

Step No.	Transformation	Formula Used
1	$ - $	Input string
2	$ - $	Formula 0
3	$ -$	Formula 0
4	$ $	Formula 1 Resultant signal

The RCNA for the above EPT system is defined by $C^{SUB} = \langle Q, INI, FIN \rangle$ where $Q = \{q_{00}, q_{10}, q_{20}, q_{11}, q_{12}, q_{21}, q_{22}, q_{30}, q_{31}, q_{32}, q_{40}, q_{41}, q_{50}, q_{51}, q_{60}, q_{61}, q_{62}, q_{70}, q_{71}, q_{72}\}$ with the set of edges T where,

$T = \{(q_{INI}, \Sigma^*, q_{00}), (q_{00}, \Sigma \setminus \xi / \leftarrow, q_{10}), (q_{10}, \Sigma \setminus \xi / \leftarrow, q_{20}), (q_{10}, 1 / \Lambda, q_{11}), (q_{11}, \mu / \rightarrow, q_{12}), (q_{12}, \Lambda / \vdash, q_{00}), (q_{20}, - / -, q_{21}),$

$(q_{21}, \mu / \rightarrow, q_{22}), (q_{22}, \Lambda / \vdash, q_{00}), (q_{30}, 1 / \Lambda, q_{31}), (q_{31}, \mu / \rightarrow, q_{32}), (q_{32}, \Lambda / \vdash, q_{00}), (q_{40}, \Sigma \setminus \xi / \leftarrow, q_{50}), (q_{40}, 1 / \Lambda, q_{41}),$

$(q_{41}, \Lambda / \vdash, q_{00}), (q_{50}, - / -, q_{51}), (q_{51}, \Lambda / \vdash, q_{00}), (q_{60}, 1 / \Lambda, q_{61}), (q_{61}, \mu / \rightarrow, q_{62}), (q_{62}, \Lambda / \vdash, q_{00}), (q_{70}, 1 / \Lambda, q_{71}), (q_{71}, - / \Lambda, q_{FIN}),$

$(q_{71}, \Lambda / \Lambda, q_{72}), (q_{72}, \Lambda / \vdash, q_{00}) \}$

The graphical representation of C^{SUB} is shown in figure 2.

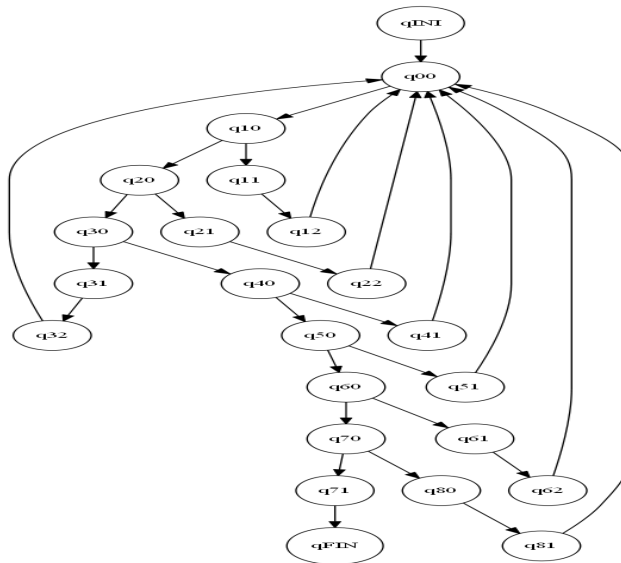


Figure 2: RCNA to Find Difference of Two Signals

PRODUCT OF TWO SIGNAL SEQUENCES

Let us consider two non negative integer sequences [3] of any two signals and then convolution of these two is nothing but multiplication of pair of integers. Let N^{MUL} is an RCNA based normal algorithm which is used to multiply to signals denoted by integer sequence. We are considering only two positive signals to make the process simple and clear. Firstly we like convert everything into sequence b's then we can apply successively formula replacing $*b$ by $|*$ which yields final result. The substitution formula for multiplication is as follows.

Table 5: Substitution Formulas for Product of Two Signals

Formula	Formula Number	Comments
$a \rightarrow ba$	0	Add $ ba$ in the place $a $ in the given input string
$b \rightarrow b$	1	Write in reverse
$O* \rightarrow O*$	2	Truncate $ $ from substring
$ *O \rightarrow *O$	3	Truncate $ $ from substring
$ * \rightarrow *a$	4	Replace $ $ by a
$* \rightarrow *$	5	Remove $ $ from string
$a \rightarrow$	6	Empty string at a
$*b \rightarrow *$	7	Replace by $ $
$* \rightarrow .$	8	$.$ in the place of $*$

For example let us consider two signals s_1, s_2 which are denoted by integer sequence x, y where $x=|||$ and $y=||$. Let us apply N^{MUL} algorithm to find product of s_1 and s_2 which is as follows.

Table 6: Product of Signals 3 and 2

Step No.	Transformation	Formula Used
1	$ * $	Input string
2	$ *a $	4
3	$ * ba $	0
4	$ * b ba$	0
5	$ * bba$	1
6	$ *a bba$	4
7	$ * ba bba$	0
8	$ * b babba$	0
9	$ * bbabba$	1
10	$*a bbabba$	4
11	$* ba bbabba$	0
12	$* b bababba$	0
13	$* bbabbabba$	1
14	$* bbabbabba$	5
15	$*bbabbabba$	5
16	$*bbbbabba$	6
17	$*bbbbbbba$	6
18	$*bbbbbbb$	6
19	$ *bbbbbb$	7
20	$ *bbbb$	7
21	$ *bbb$	7
22	$ *bb$	7
23	$ *b$	7
24	$ *$	7
25	$ $	8 (resultant string)

The RCNA for the above EPT system is defined by $C^{MUL} = \langle Q, INI, FIN \rangle$ where $Q = \{q_{00}, q_{10}, q_{20}, q_{11}, q_{12}, q_{13}, q_{30}, q_{21}, q_{22}, q_{40}, q_{50}, q_{41}, q_{42}, q_{51}, q_{52}, q_{60}, q_{70}, q_{71}, q_{72}, q_{73}, q_{80}, q_{81}, q_{82}, q_{83}, q_{90}, q_{91}, q_{92}, q_{93}, q_{100}, q_{101}, q_{102}, q_{103}, q_{104}, q_{105}, q_{110}, q_{111}, q_{112}\}$ the set of edges T where,

$$T = \{ (q_{INI}, \Sigma^*, q_{00}), (q_{00}, \Lambda, \Sigma\xi/\leftarrow, q_{10}), (q_{10}, \Sigma\xi/\leftarrow, q_{20}), (q_{11}, |/*/*a, q_{12}), (q_{12}, \mu/\rightarrow, q_{13}), (q_{13}, \Lambda/\vdash, q_{00}), (q_{20}, \Sigma\xi/\leftarrow, q_{30}), (q_{20}, a/|ba, q_{21}), (q_{21}, \mu/\rightarrow, q_{22}), (q_{22}, \Lambda/\vdash, q_{00}), (q_{30}, \Sigma\xi/\leftarrow, q_{40}), (q_{40}, \Sigma\xi/\leftarrow, q_{50}), (q_{40}, a/|ba, q_{41}), (q_{41}, \mu/\rightarrow, q_{42}), (q_{42}, \Lambda/\vdash, q_{00}), (q_{50}, b/|b, q_{51}), (q_{51}, \mu/\rightarrow, q_{52}), (q_{52}, \Lambda/\vdash, q_{00}), (q_{60}, \Sigma\xi/\leftarrow, q_{70}), (q_{70}, |/*/*a, q_{71}), (q_{71}, a/|ba, q_{72}), (q_{72}, a/|ba, q_{73}), (q_{73}, \mu/\rightarrow, q_{74}), (q_{74}, \Lambda/\vdash, q_{00}), (q_{80}, |/*/*a, q_{81}), (q_{81}, a/|ba, q_{82}), (q_{82}, a/|ba, q_{83}), (q_{83}, \mu/\rightarrow, q_{84}), (q_{84}, \Lambda/\vdash, q_{00}), (q_{90}, a/\Lambda, q_{91}), (q_{91}, a/\Lambda, q_{92}), (q_{92}, a/\Lambda, q_{93}), (q_{93}, \mu/\rightarrow, q_{94}), (q_{94}, \Lambda/\vdash, q_{00}), (q_{90}, \Sigma\xi/\leftarrow, q_{100}), (q_{100}, \Sigma\xi/\leftarrow, q_{110}), (q_{100}, *b/|*, q_{101}), (q_{101}, *b/|*, q_{102}), (q_{102}, *b/|*, q_{103}), (q_{103}, *b/|*, q_{104}), (q_{104}, \mu/\rightarrow, q_{105}), (q_{105}, \Lambda/\vdash, q_{00}), (q_{110}, *b/|*, q_{111}), (q_{111}, *b/|*, q_{FIN}), (q_{111}, \mu/\rightarrow, q_{112}), (q_{112}, \Lambda/\vdash, q_{00}) \}$$

The graphical representation of C^{MUL} is shown in figure 3

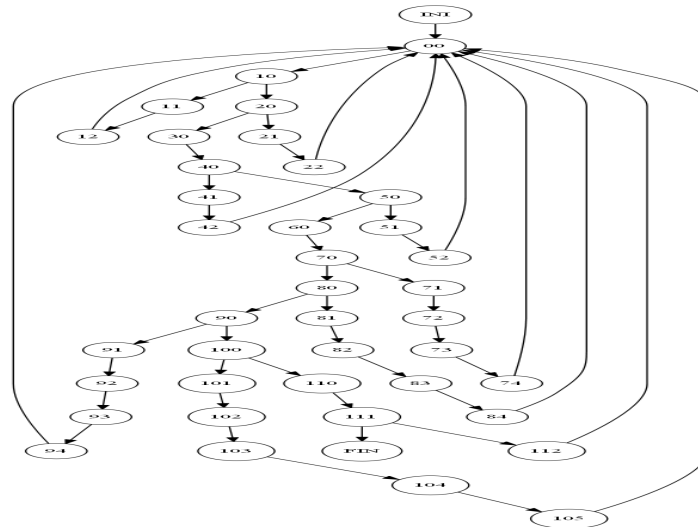


Figure 3: RCNA for Product of Signals

DIVISION OF STRING SEQUENCES

Let us consider two signals s_1, s_2 which are denoted as non negative integer sequence where s_1 is factor of s_2 . Here the division forced to remainder as zero and only for positive integers. The division of signals can be accomplished by finding the number of times s_2 occur in s_1 by factoring method [4,5].

Table 7: Normal Algorithm for Division of Signals

Formula	Formula Number	Comments
$/ * \rightarrow /*$	0	Cancel number of $ $'s in dividend which are present in divisor.
$/ \rightarrow \Lambda$	1	Replace $/$ by empty string Λ
$/ * \$ \rightarrow $	2	Write $ $ at the end of string after cancelling equal length substring from divisor and dividend. Where $\$$ is end of the string.
$\$ \rightarrow \Lambda$	3	Replace $\$$ by empty string Λ
$\vdash \rightarrow \Lambda$	4	Replace \vdash by Λ

Let us consider two signals which are represented as integer sequence $||||$ and $||$. We should use formula 0 for twice then formula 1 or once to get desired result. The division of these two signals carried out as follows.

Table 8: Division of Signals 4 and 2

Step No.	Transformation	Formula Used
1	$ / \$$	Input string
2	$ \#//\$$	0
3	$ \#//\$ $	2
4	$ \#//\$ $	0
5	$ \#//\$ $	2
6	$ \#//\$ $	4
7	$ \#//\$ $	4
8	$ \#//\$ $	4
9	$ \#//\$ $	4
10	$ \#//\$ $	1
11	$ \#//\$ $	4
12	$ \#//\$ $	4
13	$ $	3(output string)

The RCNA for the above EPT system is defined by $C^{DIV} = \langle Q, INI, FIN \rangle$ where $Q = \{q_{00}, q_{10}, q_{20}, q_{30}, q_{40}, q_{31}, q_{32}, q_{50}, q_{41}, q_{42}, q_{43}, q_{44}, q_{45}, q_{60}, q_{51}, q_{52}, q_{53}, q_{54}, q_{55}, q_{61}, q_{62}, q_{70}, q_{71}\}$ the set of edges T where,

$$T = \{(q_{INI}, \Sigma^*, q_{00}), (q_{00}, \Sigma \setminus \xi / \leftarrow, q_{10}), (q_{10}, \Sigma \setminus \xi / \leftarrow, q_{20}), (q_{20}, \Sigma \setminus \xi / \leftarrow, q_{30}), \\ (q_{30}, \Sigma \setminus \xi / \leftarrow, q_{40}), (q_{30}, ///, q_{31}), (q_{31}, \mu / \rightarrow, q_{32}), (q_{31}, \Lambda / \vdash, q_{00}), (q_{40}, \Sigma \setminus \xi / \leftarrow, q_{50}), (q_{41}, ///, q_{42}), (q_{42}, || /, q_{43}), (q_{43}, \$ / \$, q_{44}), \\ (q_{44}, \mu / \rightarrow, q_{45}), (q_{45}, \Lambda / \vdash, q_{00}), (q_{50}, \Sigma \setminus \xi / \leftarrow, q_{60}), (q_{50}, || / \#, q_{51}), (q_{51}, ///, q_{52}), (q_{52}, || / \#, q_{53}), (q_{53}, \$ / \$, q_{54}), \\ (q_{54}, \mu / \rightarrow, q_{55}), (q_{55}, \Lambda / \vdash, q_{00}), (q_{60}, \Sigma \setminus \xi / \leftarrow, q_{70}), (q_{60}, / \Lambda, q_{61}), (q_{61}, / \Lambda, q_{62}), (q_{62}, \$ / \Lambda, q_{FIN}), (q_{70}, \mu / \rightarrow, q_{71}), \\ (q_{71}, \Lambda / \vdash, q_{00}) \}$$

The graphical representation of C^{DIV} is shown in figure 4

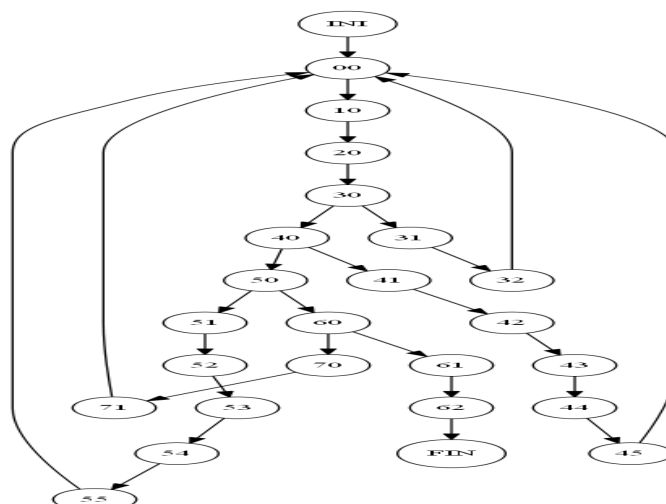


Figure 4: RCNA for Division of Signals

CONCLUSIONS

In this paper a new technique for performing operations on signals using normal algorithms and RCNA has been

represented, which solves the problems like numeric computation of signals, representation of signals in computer process able format. The use of automata in signal processing is still in infancy stage. More efforts are required to explore various methods using automata to implement various signal processing operations. The features of automata will make them as alternative for designing signal operations as conventional methods. In summary, it is considered that RCNA based automata have an excellent potential in signal processing.

REFERENCES

1. Aberth, O, Computable Analysis, McGraw Hill, New York, 1980.
2. Arbib, M. A, Algebraic Theory of Machines, Languages and Semi groups, Academic Press, New York, 1968.
3. Barwise, Jon, Handbook of Mathematical Logic, North-Holland Publishing company, 1977.
4. Beeson, M. J, Foundations of Constructive Mathematics, Springer Verlag, Heidelberg, 1985.
5. Benthem, J. F.A.K.V, The Logic of Time, D. Reidel, London, 1983.
6. Bishop, Errett, Foundations of Constructive Analysis, McGraw Hill, 1967.
7. Bobrow, D.G, Symbol Manipulation Languages and Techniques, Proceedings of the IFIP Working Conference on Symbol Manipulation Languages, North-Holland Publishing Company, Amsterdam, 1968.
8. Bridges, D. S, Constructive Functional Analysis, Pitman, London, 1979.
9. Clifford and Preston, The Algebraic Theory of Semi groups, American Mathematical Society, Providence, 1961.
10. Copi, I. M, Symbolic Logic, MacMillan Publishing Company, 1979.
11. Dalen, D. V, Logic and Structure, Springer Verlag, Heidelberg, 1980.
12. Ershov, Yu. L, and Palyutin, E.A, Mathematical Logic, Mir Publishers, Moscow, 1984.
13. Giardina, C. R, and Dougherty, E.R, Morphological Methods in Image and Signal Processing, Prentice Hall Inc, 1988.
14. Gleason, A. M, Fundamentals of Abstract Analysis, Addison Wesley. 1966.
15. Griswold, R. E, Poage, J. F, and Polonsky, I. P, The SNOBOL4 Programing Language, Prentice Hall, 1971.